EVENT BUILDER STATUS AND PLANS

13 January 2000

Stephen Tether
for the
MIT group

Contents

Intro

- Event builder architecture
- EVB proxy

Status

- Integration with Run Control
- Scaling up
- SVX calibration runs
- Problems fixed

Short-to-mid range plans

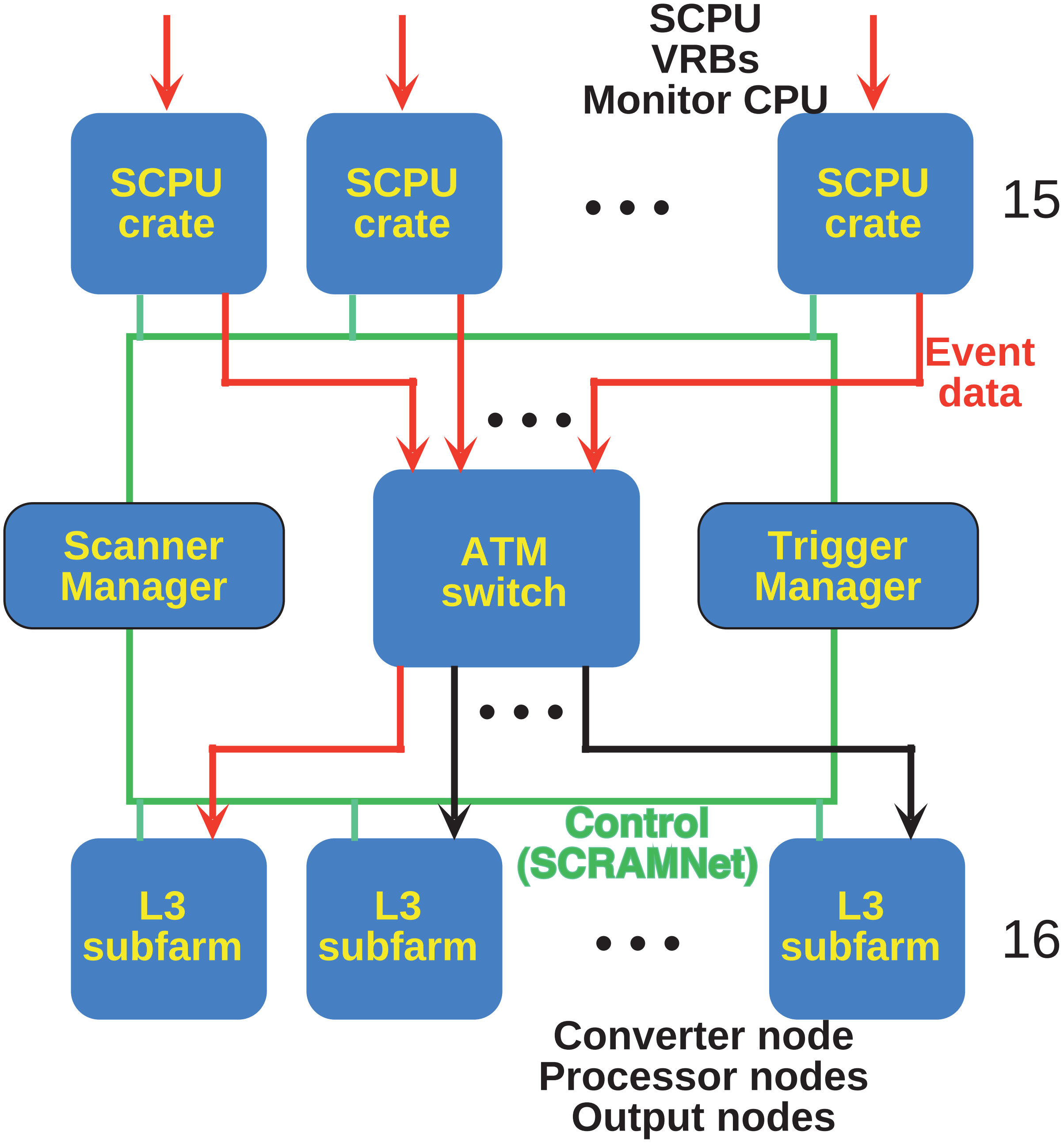
- Continue large-scale tests
- Further automation of EVB start/stop
- Expanded monitoring
- ATM software improvements
- Other

Summary

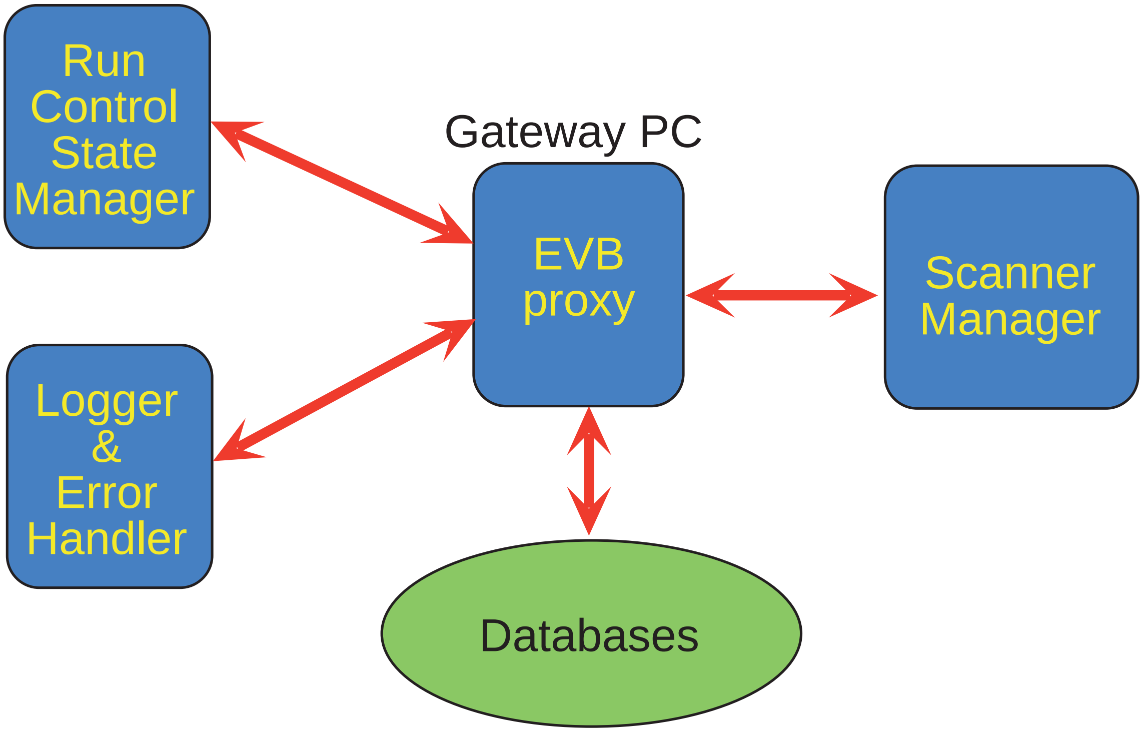
- Rough priorities
- Conclusions

Intro

Event builder architecture



The EVB Proxy



- Why the intermediate step?
  - At least for now, event builder CPUs are on a private network, so they need a gateway anyhow.
  - Keep DB and SmartSockets code off VME CPUs
  - Greater freedom of implementation, e.g., one can use Java.

Status

Integration with Run Control

Proxy

- All state transitions work
- Zephyr messages from Scanner Manager and SCPUs are relayed as Merlin simple-text messages.
- Config message implemented
- Started by hand
- Uses ASCII file to get the list of currently installed hardware
- Doesn't yet publish partition status

Scanner Manager

- Communication with the Trigger Manager works

Hardware database

- Entries describing SCPUs are complete

Scaling up

Installed

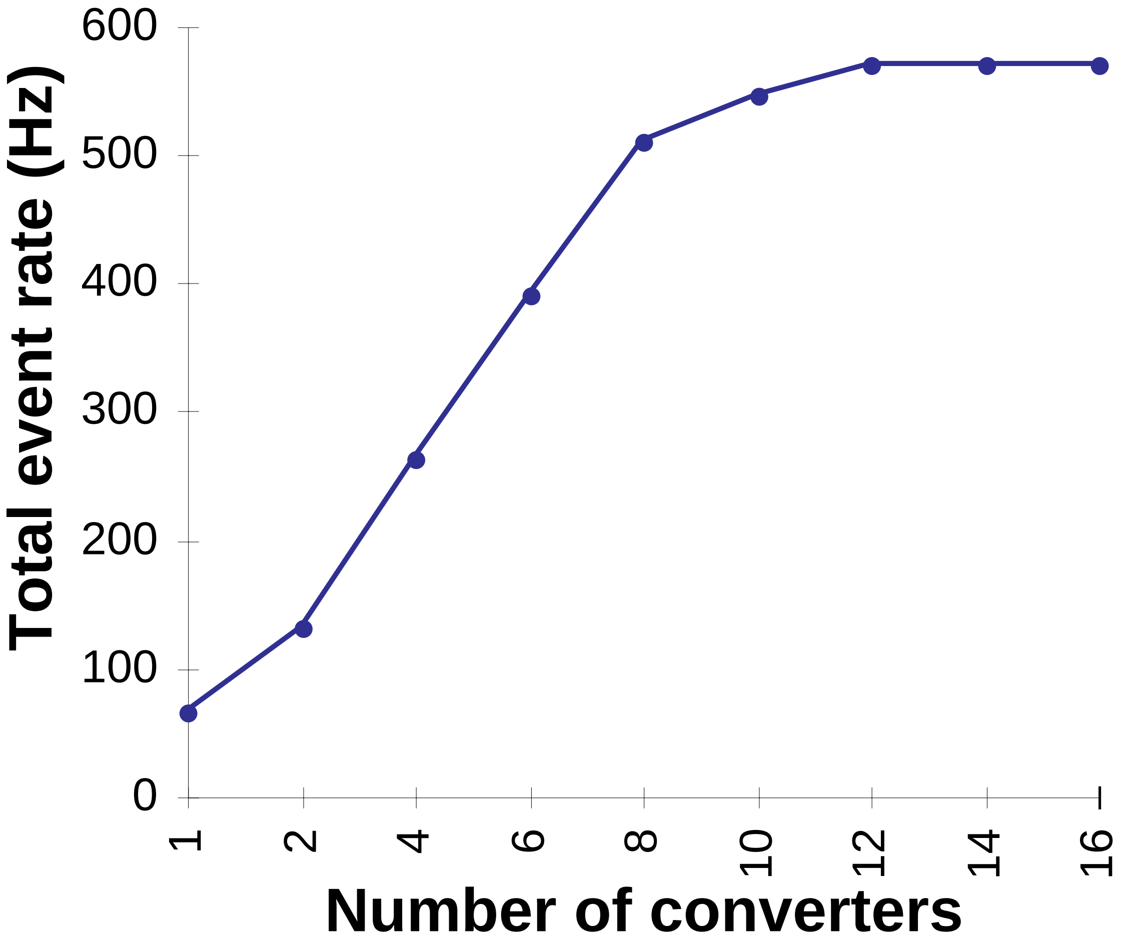
- All 15 SCPUs on 1st floor
- 16 converter nodes on 3rd floor
- SCRAMNet modules and fibers for SCPUs, SM, and converter nodes
- ATM fibers, adapters, and switch upgrade for 16x16

Throughput tests (with fake triggers)

- Fixed fragment size of 16 KB per SCPU
- Total event rate in 15 x 16 system is about 500 Hz (event size 240KB)
- Scaling in number of SCPUs is OK
- Scaling in number of converter nodes is disappointing above eight

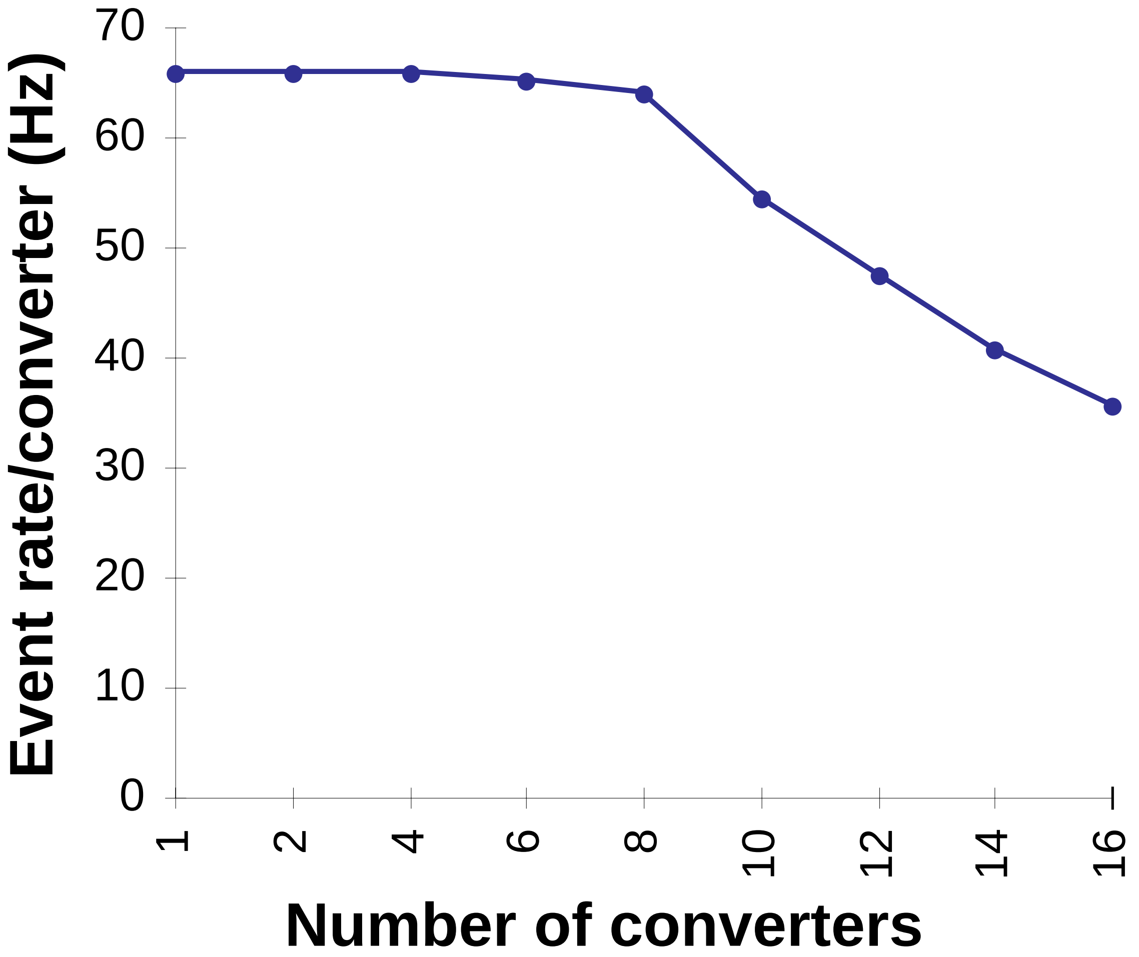
Reliability is still poor for 15x16 system running at high rates: converters gradually drop out.

1 SCPU to N converters
Total event rate



Event size = 16KB

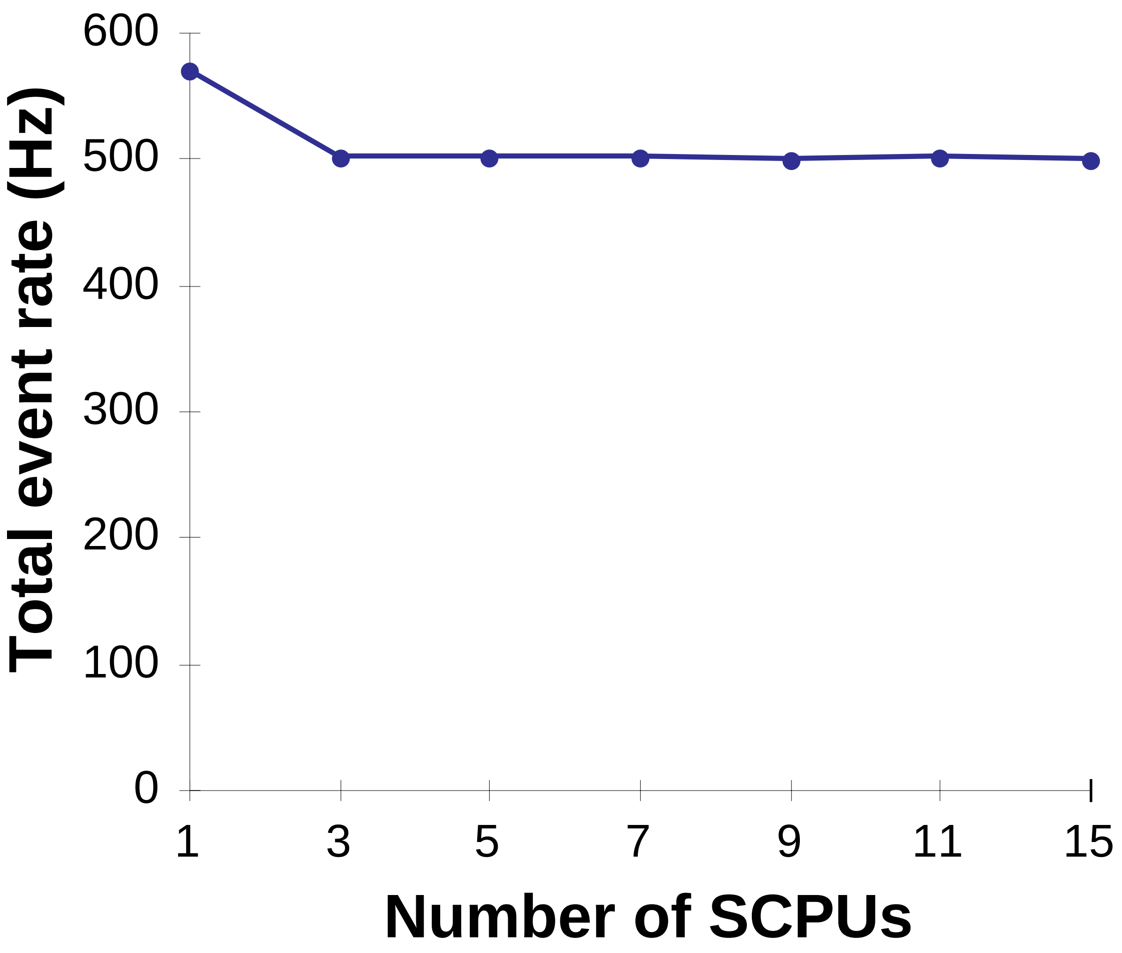
1 SCPU to N converters Event rate/converter



Event size = 16 KB

Note that converters are not at full capacity where the total event rate has leveled off. Problem in fake trigger generation???

N SCPUs to 16 converters Total event rate



Event size = 16KB * no. of SCPUs
Throughput @ 500 Hz = 117 MB/sec

SVX calibration runs

Hardware

- 1 FIB
- 1 VRB
- 1 converter node
- 1 processor node
- 1 output node
- Consumer server
- Consumer PC

Data flow

- Complete data flow from FIB to consumer works
- Consumer calculates ped. means, RMSes and stores them in database
- Some trouble yet reading them back

Problems fixed

Hardware

- Faulty SCRAMNet module replaced. Caused messages to be lost.

- Faulty SCRAMNet bypass switch replaced. Caused lost messages.

- PCI bus problems in five converter nodes from Eternal Graphics made to go away for all but one (which was sucessfully replaced). ATM card caused PC crash or SCRAMNet card was unreliable.

- All MVME2600 CPUs have working 6U-9U adapter cards and 12V power.

Problems fixed

Software

- Can now use VRBs in slots 16–20
- Limit on number of VRBs is 240, 16 per SCPU.
- PC ATM driver is now called by the kernel every 5 ms instead of being driven by device interrupts. The old way caused occasional long delays in packet processing (10–80 ms or more).
- Found a work-around for a problem with the TCP software in the version of VxWorks we were using. Simultaneous incoming messages from more than about seven converter nodes caused the Scanner Manager to run very slowly and lose connections to converter nodes.

Plans
(short-to-mid term)

Large-scale tests

Investigate 500 Hz event rate limit

Make operation more reliable

EVB automation

Proxy

- Make a permanent installation
- Implement simple stop and restart
- Have it use the hardware database

Scanner Manager and SCPUs

- Load and start their software from the startup scripts

Monitoring

SCPUs and Scanner Manager

- Release expert control software for general use

Proxy

- Add publishing of partition status at regular intervals

VRBs

- Need to look at Glink error counts and other status information for SVX VRBs
- **BUT** Reading status registers corrupts fast FIFO contents except at event boundaries
- Therefore the simplest safe way to get the info is to have the SCPU fetch it between events
- Monitoring CPU (MVME2300) needs to tell SCPU what to get and where to save the information

ATM

Node addresses

- Current scheme allows at most three event reception buffers per converter
- Removing this limit will require changes to mid-level ATM code on SCPUs, Scanner Manager, and PCs

Large fragments

- Currently, the SCPUs send all an event's data (fragment) as one packet
- Max packet size is about 59K bytes, possibly not enough for some types of calibrations
- Need to implement multi-packet sending and receiving for fragments

Other

Send back faulty hardware

Get better TCP stack for VxWorks

- Use later VxWorks BSP or install newer TCP software

Install serial-port concentrator

- For VxWorks and Linux machines

Change partition numbering

- Scanner Manager and SCPUs use 1–8 internally (from Run Ib).

- Proxy and L3 code add or subtract one where approriate

- Change to consistent numbering everywhere

Summary

Rough priorities

1) RC integration and EVB automation

2) Large-scale operations at high rates

3) Transmission of large fragments

4) VRB monitoring

5) Everything else

Conclusions

Very first rate tests of 15x16 system top at about 500 Hz. Not bad, but it can probably do better.

Still have to fix reliability problems at high-est rates for this large a system.

RC integration almost done.